Multi-University Research Initiative on
High-Confidence Design for Distributed Embedded Systems

Frameworks and Tools for High-Confidence Design of Adaptive, Distributed Embedded Control Systems

Year 3 Progress Report

Vanderbilt University
Institute for Software Integrated Systems
2015 Terrace Place, Nashville, TN 37203
(615) 322-3455 (office)
(615) 343-7440 (fax)
janos.sztipanovits@vanderbilt.edu

TEAM MEMBERS:

Vanderbilt: J. Sztipanovits (PI) and G. Karsai
UC Berkeley: C. Tomlin (Lead and co-PI), Edward Lee and S. Sastry
CMU: Bruce Krogh (Lead and co-PI) and Edmund Clarke
Stanford: Stephen Boyd

FA9550-06-0312

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2009		**00-00-2009 to 00-00-2009**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Frameworks and Tools for High-Confidence Design of Adaptive, Distributed Embedded Control Systems	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Vanderbilt University,Institute for Software Integrated Systems,2015 Terrace Place,Nashville,TN,37203**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**21**	

1. Objectives

This project aims to develop a comprehensive approach to the model-based design of high-confidence distributed embedded systems. We will take advantage and fully leverage a shared theoretical foundation and technology infrastructure in four focus areas: hybrid and embedded systems theory, model-based software design, composable tool architectures and experimental testbeds. The objectives of our research in the focus areas are the following:

1. Develop theory of deep composition of hybrid systems with attributes of computational and communication platforms. We will address compositionality, concurrency, heterogeneity and resource, robustness, approximate verification and adaptive control architectures for uncertainty handling.
2. Develop foundations of model-based software design for high-confidence, networked embedded systems applications. We will investigate new semantic foundations for modeling languages and model transformations, precisely architected software and systems platforms that guarantee system properties via construction, and new methods for static source code verification and testing, as well as for dynamic runtime verification and testing.
3. Develop composable tool architecture that supports high-level reusability of modeling, model-analysis, verification and testing tools in domain-specific tool chains. We create new foundation for tool integration that goes beyond data modeling and data transfer.
4. Demonstrate the overall effort by creating an end-to-end design tool chain prototype for the model-based generation and verification of embedded controller code for experimental platforms.

2. Status of the Effort

We have reached the following major milestones toward the compositional design of high confidence embedded control systems on computational and communication platforms.

1. We have achieved new results in hybrid control system design using reachable set analysis: a methodology for computing reachable sets using quantized inputs over discrete time steps has been developed and implemented for an aircraft collision avoidance example. We have used reachable set analysis in complex control law design, and have demonstrated its use (in simulation) on aerobatic maneuver design for the STARMAC quadrotor helicopter testbed. In related work, we have developed a new optimization scheme for scheduling hybrid systems, and have demonstrated the results on an autonomous car simulation testbed. We are focusing efforts this summer for both projects in demonstration of the algorithms on the actual testbeds.
2. We have extended our approach for integrated software model checking in the loop to the case of nonlinear dynamic plant models using the concept of bisimulation functions for nonlinear systems.
3. We developed a new widening operator for verification of numerical programs that is much less conservative than standard widening operators used to accelerate the termination of fixed point computations in abstract interpretation. We initiated work on ar-

chitecture-level tools for modeling and verifying properties of embedded system design specifications early in the design process.

4. We have continued developing the passivity based approach for networked controller design and demonstrated feasibility experimentally.
5. We have completed the working prototype of an end-to-end tool chain for the model-based design of networked control systems. The toolchain integrates a verification step to verify code 'as running on the physical platform'. The underlying implementation platform is the Time-Triggered Architecture (TTA), realized on two processor types and communication buses. We have built demonstrations for auto-generating code from verified models.
6. Translating models into efficient executable embedded code that reliably implements the model semantics continues to be a challenging problem. We have re-architected the Ptolemy II code generation infrastructure to provide an adaptable and extensible platform that supports experimentation with code generation for a variety of models of computation and target platforms. We are using and extending this framework to build code generators based on synchronous dataflow, finite state machine, synchronous/reactive, Giotto, and Ptides models of computation, and have shown that we can target bare-iron microcontrollers, lightweight microkernels, and real-time operating systems.

3. Accomplishments and New Findings

We continued our work on developing tools, methods and other components of the project along the four objectives.

3.1 Hybrid and Embedded Systems Theory

3.1.1 Embedded Systems Modeling and Deep Compositionality (Krogh, Tomlin, Sastry)

While our progress in previous years has focused on the computation and use of reachable sets for simple protocol verification and analysis, over this past year we have begun to develop a method for hybrid system trajectory planning, using the high level mode description. Related research efforts in the past have developed symbolic languages for robot motion planning, for describing complex system behavior, or estimating behavior from observation, this is the first time that the broad concept of reachable sets has been used to provide *a priori* verified behavior, at the planning level. We have demonstrated the technique in simulation on a challenging problem (aerobatic maneuvers for our STARMAC quadrotor platform), and we are currently working on the actual flight implementation.

3.1.2 Hierarchies of Robust Hybrid and Embedded Systems (Tomlin, Krogh, Sastry)

Reachability analysis. We have extended our technology for reachable set design to include quantized input signals, and discrete time implementation. This makes the technique much more useful in practice. We have implemented this extension on simulation examples of collision avoidance for two civilian aircraft, and an automated re-fueling example involving a UAV and a large tanker. We have also developed methods for path planning around reachable sets: using a

combination of forward reachable set computation, and convex underapproximation of the region around reachable sets, we have shown how the trajectory planning method can be posed as a convex optimization program.

Optimization of hybrid systems. We have developed an optimal control algorithm for switched hybrid systems, which given a cost function and a set of inequality constraints computes the optimal sequence of discrete states, the optimal times for switches between discrete states, and the optimal continuous input for each state. Our method is iterative, involving updates on the optimal mode switch time by perturbing an initial switch schedule, therefore the results from our method are only locally optimal. We have implemented this scheme in simulation on an automated car testbed, using our method as a trajectory planner, and we expect to use this algorithm in a real car in the future.

3.1.3 Constructive Architectures for Digital Controllers of Continuous Time Systems (Kottenstette)

Using passivity and scattering theory we have shown how to interconnect either a linear or non-linear passive continuous time system to a passive digital controller in which continuous time stability (L_m^2-stability) can be guaranteed. The key to creating such a system is to transform the continuous time input-output signals of the plant to wave-variables. The continuous-time wave variables are then interfaced to a passive sampler (PS) which converts a continuous-time-wave-variable to a discrete-time-wave-variable in a causal manner at an *arbitrary* sampling rate T_s, analogously a passive hold (PH) converts a discrete-time-wave-variable to a continuous-time-wave-variable in a passive manner at an update rate T_s. These discrete-time wave-variables can then be transmitted over a digital network and in-spite fixed-time delays and data-dropouts of the wave-variables L_m^2-stability can be preserved. In fact, time delays, such as those incurred over TCP/IP communication networks, can be tolerated without any modification to the original system. Other communications protocols such as UDP can also be used as long as duplicate wave-variable transmissions are dropped. We have successfully applied this architecture to both linear and non-linear systems including robotic-arm-manipulators. In order to show L_m^2-stability, an important analysis tool, which we refer to as the *inner-product-equivalent-sampler and zero-order-hold* (IPESH), is used to relate non-wave-continuous-time variables to non-wave-discrete-time variables. In fact, the *IPESH-transform* now makes it possible to synthesize discrete-time linear-time-invariant controllers which closely match their continuous-time counterpart in both magnitude and phase response up to the Nyquist frequency (π/T_s) without any need for 'pre-warping'. The IPESH-transform, makes it extremely easy to generate a digital controller which can cancel out non-ideal lag effects such as those encountered by the rotor-angular-velocity to rotor-thrust characteristics associated with quad-rotor aircraft. This passivity based frame-work has been successfully applied to the control of multiple-plants and controllers. In particular we have shown how a *power junction* can be used to interconnect multiple-plants and controllers while preserving system stability. The averaging-power junction has been shown to allow multiple continuous-time plants with the same steady-state gain (yet different dynamic transients) to be commanded by a single PID-digital-controller to track a given trajectory. Furthermore, through the use of a *resilient power junction* we have shown how a digital control network can be constructed in which redundant passive-digital controllers can be lost and even corrupted while preserving over-all system stability.

Using conic-systems theory, a more generalized form of passivity theory, we have shown how stabilizing controllers for systems consisting of cascades of passive-sub-systems such as the quad-rotor aircraft can be constructed. In fact, the overall control-architecture essentially consists of nested diagonal proportional feedback control loops and a single saturation block to account for actuator saturation limits. In spite of the broad applicability to both linear and non-linear systems, the advantages of the control-architecture include being easy to construct and understand, requires a minimal-amount of mathematical operations, and possess significant robustness to system uncertainty. For example, in practice, we have found our architecture to be quite insensitive to modest sampling rate and delay even when applied to the control of non-linear systems such as the quad-rotor aircraft

3.1.4 Verification and Validation of Conservative Approximations (Clarke, Krogh)

Bounded-time verification technique that combines software model checking and simulation. We extended our method for integrating source-code model checking with dynamic system analysis to verify properties of controllers for nonlinear dynamic systems. Source-code model checking verifies the correctness of control systems including features that are introduced by the software implementation, such as concurrency and task interleaving. Sets of reachable continuous state variables are computed using numerical simulation and bisimulation functions. The technique as originally proposed handles stable dynamic systems with affine state equations for which quadratic bisimulation functions can be computed easily. The extension in this past year handles nonlinear systems with polynomial state equations for which bisimulation functions can be computed in some cases using sum-of-squares (SoS) techniques. The algorithm includes the convex optimizations required to perform control system verification using a source-code model checker, and the method is illustrated for an example of a supervisory control system.

Systematic search for counterexamples using model checking and numerical simulation. We extended the trajectory sensitivity work to a parameter synthesis problem for nonlinear hybrid systems. Considering a set of uncertain parameters and a safety property, we give an algorithm that returns a partition of the set of parameters into subsets classified as safe, unsafe, or uncertain, depending on whether respectively all, none, or some of their behaviors satisfy the safety property. We make use of sensitivity analysis to compute approximations of reachable sets and an error control mechanism to determine the size of the partition elements in order to obtain the desired precision. We apply the technique to Simulink models by combining generated code with a numerical solver that can compute sensitivities to parameter variations. We present experimental results on a non-trivial Simulink model of a quadrotor helicopter.

3.1.5 Statistical Probabilistic Model Checking (Clarke, Platzer)

Stochastic systems arise naturally, for example, because of uncertainties present in a system's environment (e.g., the reliability of communication links in a wireless sensor network, the rate of message arrivals on an aircraft's communication bus, or the number of contenting peers in a Bluetooth device discovery phase). Uncertainty is usually modeled via a probability distribution, thereby resulting in stochastic systems, i.e., systems which exhibit probabilistic behavior. These are clearly very important systems with many practical applications, which motivated our investigation on how Model Checking can be applied to stochastic systems. The problem of Model Checking stochastic systems is quite different from the Model Checking of standard systems. Because of the probabilistic behavior, one has to introduce a notion of probability in the concept

"the system satisfies a specification". Suppose we are given a temporal logic formula f and a closed (i.e., with no free parameters or inputs) stochastic system M. Then we can assign a unique probability p to the event "system M satisfies property f". The following question is now well-posed: is p greater (or smaller) than t (where t is a user-defined threshold probability, which in general depends on the property being verified)? The Probabilistic Model Checking problem thus amounts to finding out whether a system satisfies a specification with at least (or at most) a fixed probability. For example: "does the system fulfill a request within 1ms with probability at least 0.99?"

Numerical methods solve the Probabilistic Model Checking problem by first computing the (unknown) probability p and then comparing it with the threshold t. However, these methods do not scale up to realistic systems. Our solution to the Probabilistic Model Checking problem is instead based on randomized sampling of the system's traces and statistical hypothesis testing. Therefore, the statistical conclusion is not guaranteed to be correct, but the probability of giving a wrong answer is bounded. The benefit of our approach is that a conclusion is often reached significantly faster than with numerical techniques, thus making the approach more scalable to the challenging size and complexity of our target systems.

We have successfully applied our approach to a model of a Delta-Sigma modulator for which previous formal verification attempts were too conservative and required excessive computation time. We have also started investigating the use of Statistical Probabilistic Model Checking for the verification of Stateflow-Simulink models with a hybrid dynamics. In particular, we have developed a new algorithm for solving the Probabilistic Model Checking problem. The algorithm uses a statistical sequential approach based on Bayes's theorem. The sequential character of our approach means that the number of sampled traces is not fixed *a priori*, but it is instead determined at "run-time". The use of Bayes's theorem enables our algorithm to take advantage of previous knowledge about the model, where available. We have showed that, on several representative examples, our algorithm generally leads to faster verification than state-of-the-art approaches, based on either statistical or numerical techniques. These very encouraging results show that Model Checking techniques are likely to scale up to real-world hybrid systems.

3.1.6 Verification of Hybrid Systems via Differential Invariants (Clarke, Platzer)

In air traffic control, collision avoidance maneuvers are used to resolve conflicting flight paths that arise during free flight. Aircraft collision avoidance maneuvers are important and complex applications. Several maneuvers have been proposed already that assume instant turns in mid flight. Real aircraft, however, can only follow sufficiently smooth flyable curves. Hence, mathematical maneuvers that require instant turns give physically impossible conflict resolution advice. Yet curved flight exhibits nontrivial continuous behavior. In combination with the control choices during air traffic maneuvers, this yields hybrid systems with challenging interactions of discrete and continuous dynamics.

As a case study illustrating the use of a new proof assistant for a logic for nonlinear hybrid systems, we have analyzed collision freedom of roundabout maneuvers in air traffic control. In this domain, appropriate curved flight, good timing, and compatible maneuvering are crucial for guaranteeing safe spatial separation of aircraft throughout their flight. We have shown that formal verification of hybrid systems can scale up to curved flight maneuvers required in aircraft control applications.

We have introduced a fully flyable variant of the roundabout collision avoidance maneuver and verified safety properties by compositional verification. In contrast to other approaches, we verify the hybrid system dynamics without solving the differential equations and without numer-

ical errors. We use a continuous generalization of induction, for which our algorithm computes the required differential invariants. As a means for combining local differential invariants into global system invariants in a sound way, our fixed-point algorithm works with a compositional verification logic for hybrid systems. By complementing our symbolic verification algorithm with a robust version of numerical falsification, we obtain a fast and sound verification procedure, and achieve better automation. These results indicate that hybrid systems are a promising direction of further research for the aviation domain, including unmanned aerial vehicles

3.2 Model-Based Software Design and Verification

3.2.1 Model-Integrated Computing (Sztipanovits, Karsai, Kottenstette)

Cross-layer abstractions. Model - based software design progresses along abstraction layers (design platforms) capturing essential design concerns. Effectiveness of the model-based design largely depends on how much the design concerns (captured in the abstraction layers) are orthogonal, i.e., how much the design decisions in the different layers are independent. Heterogeneity of embedded systems causes major difficulties in this regard. The controller dynamics is typically designed without considering implementation side effects (e.g. numeric accuracy of computational components, timing accuracy caused by shared resource and schedulers, time varying delays caused by network effects, etc.). Compositionality in one layer depends on a web of assumptions to be satisfied by other layers.

We have continued investigating theories and techniques for applying cross-layer abstraction to make the controller designs robust against implementation side effects. We pursue this by inserting implementation related abstractions in the controller design, and physical abstractions in software design. The ultimate goal is decreasing the entanglement across the design layers.

We have developed model transformation tools that generate TrueTime abstractions from system level models and investigate now the application of orthogonal structures in implementing dynamics.

3.2.2 Autocoding Embedded software for Safety Critical Systems (Lee)

Professor Lee's group at Berkeley has furthered the development of semantic-preserving code generation in two areas: (1) code generation of Giotto models using the Precision Timed (PRET) Architecture and (2) Compositional Code Generation.

As part of her Master's Thesis, Shanna-Shaye Forbes developed code generation from the Giotto model of computation in Ptolemy II to the Precision Timed (PRET) architecture.

- Giotto is a programming model for embedded control systems that is applicable to hard real-time specifications that are periodic and features multi-modal behavior. Examples of such systems include fly-by-wire or brake-by-wire systems where sensor readings must be periodic and there are multiple modes of operation.
- PRET is a computer architecture that emphasizes predictable timing.
- Ptolemy II is an open source modeling and simulation framework that supports model-based design, and facilitates actor oriented and objected oriented programming. It serves as a laboratory for the modeling and simulation necessary in the design of a real-time embedded system. Ptolemy II has an implementation of the Giotto programming model that allows the simulation of Giotto models in Ptolemy II. Ptolemy

II also has an extensible C code generation framework can be retargeted for different targets.

In our code generation approach, we employ the correct-by-construction premise and make use of PRET's deadline capabilities to generate C code which fulfills the timing constraints of the model. We demonstrated our facility by running the generated code on the cycle-accurate PRET simulator which lets us verify that our designs meet the hard real-time deadlines.

In addition to generating PRET code, we retargeted our Giotto code generation to OpenR-TOS, a real-time operating system for embedded platforms. To illustrate these techniques we extend the code generation framework within Ptolemy II to generate C code for the Giotto programming model. We have implemented a C code generation adapter in Ptolemy II for the Giotto model of computation targeted to systems capable of running the OpenRTOS operating system. We presented an elevator controller as an example that uses the code generation framework.

Bert Rodiers, Jackie Man-kit Leung and others have been developing Compositional Code Generation, which we define as the act of automatically generating code on a per composite actor basis. By generating code in this manner, we hope to be able to compose generated code with interpreted code in simulation, which will result in performance improvements for high performance models. Compositional Code Generation will also allow us to reuse auto generated code without regenerating it each time. Similar to the spirit of co-simulation, where subsystems are executed modularly, compositional code generation combined with semantic preservation allow us to retarget submodels to different contexts. This work is in progress, we are studying the modularity problem by advancing interface theories to describe our simulation and code generation components. We are also refactoring our earlier code generation system and taking advantage of lessons learned, such as removing model of computation specific details from the code generation kernel.

3.2.3 Automated Source Code Verification and Testing (Clarke, Platzer, Krogh)

Verification of numerical code. Verification of numerical code is an important problem in embedded systems since computational artifacts such as overflow, underflow, error accumulation, divide by zero, etc., which are not present in the idealized models used for algorithm design, can lead to unexpected and possibly catastrophic consequences in many applications. We developed a static analysis technique for polyhedral domains to compute bounds on the variables in numerical code with linear arithmetic, and introduce a new widening operator that can be more precise than standard widening for iterative computations. We also developed heuristics for reducing the complexity of the analysis.

3.3 Composable Tool Architectures

3.3.1 Advanced Open Tool Integration Framework (Karsai, Sztipanovits)

Formal specification of behavioral semantics. We have continued our efforts on the formal specification of behavioral semantics for domain specific modeling languages. In the last year we have started examining Sifakis' Behavior-Interaction-Priority (BIP) model as an abstraction layer in the design flow.

3.3.2 Prototype Tool Chain (Volgyesi, Karsai, Sztipanovits)

Prototype toolchain. We continued our work on the prototype tool chain, based on the modeling language ESMoL. The architecture of the tool chain is show in Figure 1, while Figure 2 shows the specific tools and logical flow across the tools. The tool chain is capable to work with high level (controller) models imported from the MATLAB/Simulink environment (**MDL2MGA** tool), partition and assign components to nodes and tasks (ESMoL domain specific modeling language and the GME modeling environment) and generate code and runtime configuration for different distributed platforms (TTP/C, Linux, FreeRTOS). The code is generated in two steps; first the abstract syntax tree of the code is built (**SL/SF CodeGen** tools, SFC domain specific modeling language), then the actual C/C++ (optionally: Java) code is printed from the abstract model (**SFCPrint** tool). The most important benefits are the relatively low cost of adding support for additional programming languages and high level access to the executable code for external tools (e.g. source code verification)

The experimental platform includes a Linux-based TTA realization (the FRODO TTA virtual machine, running on a low-end Linux board, called the Gumstix platform) and a RTOS-based TTA realization (the same virtual machine, running on a low-end microcontroller board, called the Robostix platform). The same hardware platforms are used in the Stanford STARMAC vehicle. During the past years we have developed the platform-specific code generators that produce integration code for these platforms (the functional code generators were implemented in previous years). We have developed the I/O drivers for the low-end controller that work with the time-triggered run-time scheduler. We have extended the design-time, offline scheduler to schedule the periodic message transfers on the shared bus, and to account for I/O overhead in the scheduling.

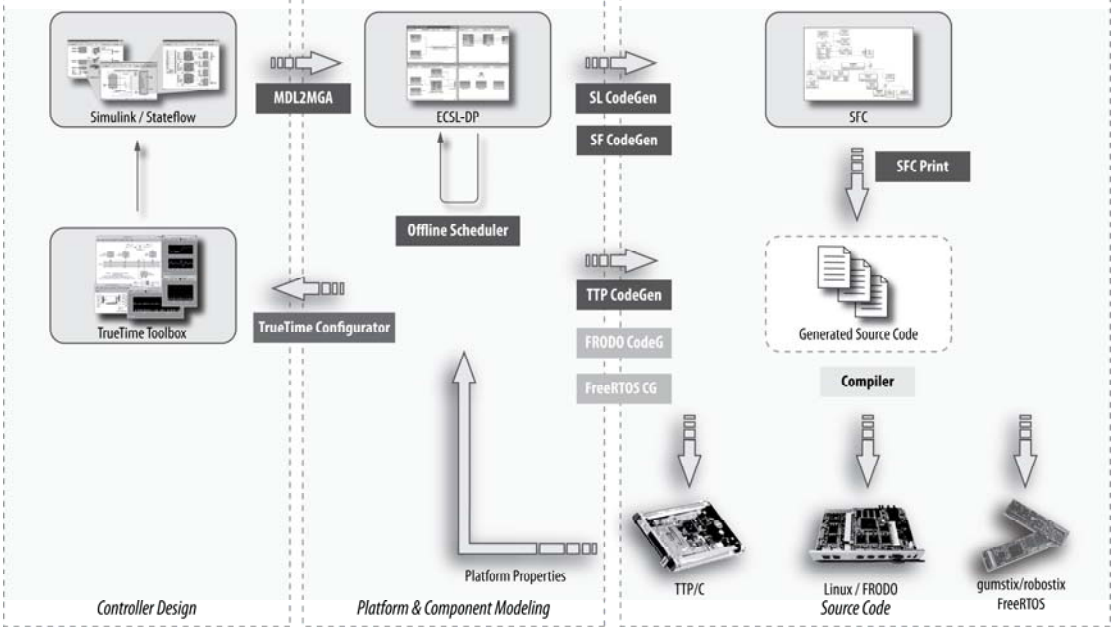

Figure 1: Architecture of the tool chain

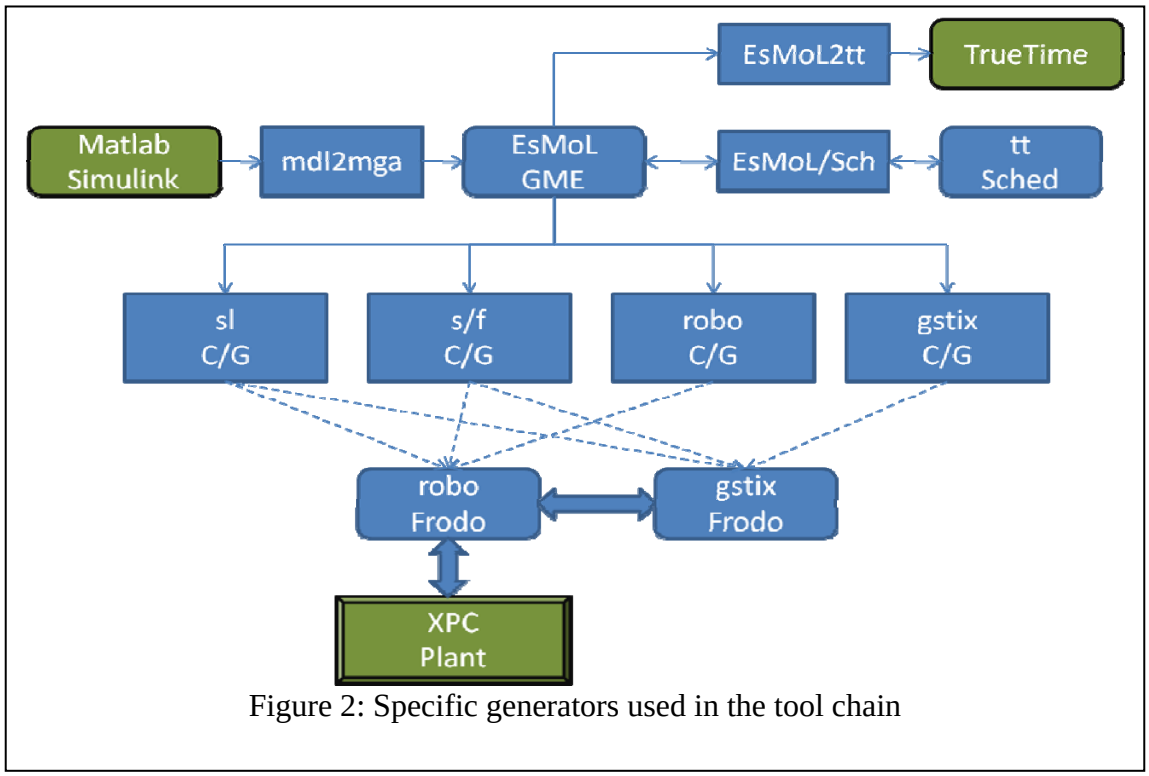

Figure 2: Specific generators used in the tool chain

We have revised the integration of the TrueTime simulation and verification toolbox with the tool chain. This integration now allows the high-fidelity simulation of the control system together with the plant, such that all platform effects (such as delays caused by message transfers and scheduling jitter) are faithfully included. Once the controller models are componentized, and the component deployment is modeled, the offline scheduling tool produces a feasible schedule. This schedule is then used to configure the TrueTime simulation, such that platform effects in the closed loop control could be studied. Observations on this simulation provide valuable feedback for the designer on how well the actual controller implementation will work in the real environment. Once the design is found satisfactory, the code can be generated and deployed on the real platform (robo+gstix) and its performance studied using a Hardware-in-the-Loop simulation (utilizing the XPC platform from Mathworks).

3.4 Testing and Experimental Validation (Tomlin, Sastry, Lee, Karsai)

We continued testing the baseline controller design of the UAV platforms on the emerging model-based design tool suite.

We have finished the construction of the real-time simulation environment for the Stanford STARMAC quadrotor aircraft control software, although the testbed architecture can support arbitrary plant models (using the Mathworks/xPC tools) and hardware controllers (currently using the Robostix+Gumstix pair). The interface between the plant simulator and the controller is ‘hard real-time’, and the xPC box simulates the real-time behavior of the plant with high-fidelity (e.g., inner loop control can be easily run at 100Hz). The control software is generated and configured with the tool chain.

4. Personnel Supported

Vanderbilt:

1. Professor Janos Sztipanovits (PI)
2. Professor Gabor Karsai
3. Nicholas Kottenstette (research scientist)
4. Joe Porter (Graduate Student, funded by this contract)
5. Graham Hemingway (Graduate Student, partially funded by this contract)
6. Ryan Thibodeaux (Graduate Student, funded elsewhere)

Associated but not supported:

1. Himansu Neema (Senior Engineer)
2. Sandeep Neema (Senior Research Scientist)
3. Harmon Nine (Senior Engineer)

Berkeley:

1. Professor Claire Tomlin
2. Professor Edward A. Lee (Faculty, funded elsewhere)
3. Professor Shankar Sastry
4. Jerry Ding (Graduate student, funded by this contract)
5. Justin Hsia (Graduate student, funded by this contract)
6. Humberto Gonzales (Graduate student, funded by this contract)
7. Shanna-Shaye Forbes (Graduate student, funded elsewhere)
8. Man-kit (Jackie) Leung (Research staff, funded by this contract)
9. Christopher Brooks (Software Engineer, funded 25%)
10. Bert Rodiers (Graduate student, funded elsewhere)
11. Haomiao Huang (Stanford graduate student, funded elsewhere)
12. Jeremy Gillula (Stanford graduate student, funded elsewhere)
13. Michael Vitus (Stanford graduate student, funded elsewhere)

CMU

1. Professor Bruce Krogh
2. Professor Edmund Clarke
3. James Kapinski, Post Doc, Dept. of ECE, CMU
4. Akshay Rahjans, Ph.D. candidate, Dept. of ECE, CMU
5. Ajinkya Y. Bhave, PhD candidate, Dept. of ECE, CMU
6. Alexandre Donzé, Post Doc, Computer Science Dept., CMU

Associated but not supported:

1. André Platzer, Professor, Computer Science Dept., CMU
2. Axel Legay, Post Doc, Computer Science Dept., CMU
3. Azadeh Farzan, Post Doc, Computer Science Dept., CMU
4. Himanshu Jain, PhD candidate, Computer Science Dept., CMU
5. Sumit Jha, PhD candidate, Computer Science Dept., CMU
6. Stephen Magill, PhD candidate, Computer Science Dept., CMU
7. Bryant Lee, PhD candidate, Computer Science Dept., CMU
8. Nishant Sinha, PhD candidate, Computer Science Dept., CMU
9. Constantinos Bartzis, Post Doc, Computer Science Dept., CMU
10. Tamir Heyman, Post Doc, Computer Science Dept., CMU
11. Azideh Farzan, Post Doc, Computer Science Dept., CMU
12. Silke Wagner, Post Doc, Computer Science Dept., CMU
13. Alexandre Donze, Post Doc, Computer Science Dept., CMU
14. Ingo Feinerer, Visiting Researcher, Computer Science Dept., CMU

Stanford

1. Professor Stephen P. Boyd,
2. Jo¨elle Skaf, Ph.D. Candidate
3. Siddharth Joshi, Ph.D. Candidate
4. Almir Mutapcic, Ph.D. Candidate
5. Seung Jean Kim, Consulting Professor

5. Publications

1. Flavio Lerda, James Kapinski, Hitashyam Maka, Edmund M. Clarke, and Bruce H. Krogh, Model checking in-the-loop, 2008 American Control Conference, Seattle, June 2008.
2. Ajinkya Y. Bhave and Bruce H. Krogh, Performance Bounds on State-Feedback Controllers with Network Delay, IEEE Conference on Decision and Control, Dec. 2008
3. James Kapinski,, Alexandre Donze, Flavio Lerda, Hitashyam Maka, Silke Wagner, and Bruce H. Krogh, Control Software Model Checking Using Bisimulation Functions for Nonlinear Systems, IEEE Conference on Decision and Control, Dec. 2008
4. Alexandre Donzé, Bruce Krogh, Akshay Rajhans, Parameter Synthesis for Hybrid Systems with an Application to Simulink Models, Hybrid Systems: Computation and Control, San Francisco, April 2009.
5. Hitashyam Maka, Goran Frehse, Bruce H. Krogh, Polyhedral Domains and Widening for Verification of Numerical Programs, Workshop on Verification of Numerical Software, San Francisco, April 2009.
6. André Platzer, Edmund M. Clarke: Computing Differential Invariants of Hybrid Systems as Fixedpoints. Formal Methods in System Design, *to appear*
7. Edmund M. Clarke, Alexandre Donzé, Axel Legay: Statistical Model Checking of Mixed-Analog Circuits with an Application to a Third Order Delta-Sigma Modulator. Formal Methods in System Design, *to appear*

8. Yu-Fang Chen, Azadeh Farzan, Edmund M. Clarke, Yih-Kuen Tsay, Bow-Yaw Wang: Learning Minimal Separating DFA's for Compositional Verification. TACAS 2009: 31-45
9. André Platzer, Edmund M. Clarke: Computing Differential Invariants of Hybrid Systems as Fixedpoints. CAV 2008: 176-189
10. Himanshu Jain, Daniel Kroening, Natasha Sharygina, Edmund M. Clarke: Word-Level Predicate-Abstraction and Refinement Techniques for Verifying RTL Verilog. IEEE Trans. on CAD of Integrated Circuits and Systems 27(2): 366-379 (2008)
11. Xenofon Koutsoukos, Nicholas Kottenstette, Joe Hall, Panos Antsaklis, Janos Sztipanovits , Passivity-Based Control Design of Cyber-Physical Systems", *Proceedings of the Workshop on Cyber-Physical Systems - Challenges and Applications (CPS-CA'08)* held in conjunction with In conjunction with the 4th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS'08)
12. Kottenstette, N., X. Koutsoukos, J. Hall, P. J. Antsaklis, and J. Sztipanovits, "Passivity-Based Design of Wireless Networked Control Systems for Robustness To Time-Varying Delays", 29th IEEE Real-Time Systems Symposium (RTSS 2008), Barcelona, Spain, IEEE, pp. 15-24, 12/2008.
13. Kottenstette, N., and J. Porter, "Digital Passive Attitude and Altitude Control Schemes for Quadrotor Aircraft", Technical Report, Nashville, TN, Institute for Software Integrated Systems, Vanderbilt University, pp. 1-12, 11/2008.
14. Kottenstette, N., X. Koutsoukos, J. Hall, J. Sztipanovits, and P. J. Antsaklis, "Passivity-Based Design of Wireless Networked Control Systems Subject To Time-Varying Delays", Technical Report, Nashville, TN, Institute for Software Integrated Systems, Vanderbilt University, pp. 1-17, 08/2008.
15. Kottenstette, N., and P. J. Antsaklis, "Wireless control of passive systems subject to actuator constraints", 47th IEEE Conference on Decision and Control (CDC 2008), Cancun, Mexico, IEEE, pp. 2979-2984, 12/2008.
16. Kottenstette, N., and P. J. Antsaklis, "Wireless Digital Control of Continuous Passive Plants Over Token Ring Networks", International Journal of Robust and Nonlinear Control: Special Issue on Control with Limited Information, 11/2008.
17. Porter, J., Z. Lattmann, G. Hemingway, N. Mahadevan, S. Neema, H. Nine, N. Kottenstette, P. Volgyesi, G. Karsai, and J. Sztipanovits, "The ESMoL Modeling Language and Tools for Synthesizing and Simulating Real-Time Embedded Systems", 15th IEEE Real-Time and Embedded Technology and Applications Symposium, San Francisco, CA, 04/2009.
18. Kottenstette, N., and N. Chopra, "Lm2-stable digital-control networks for multiple continuous passive plants", Technical Report, Nashville, TN, Institute for Software Integrated Systems, Vanderbilt University, pp. 1-14, 04/2009.
19. Eyisi, E., J. Porter, J. Hall, N. Kottenstette, X. Koutsoukos, and J. Sztipanovits, PaNeCS: A Modeling Language for Passivity-based Design of Networked Control Systems, , Nashville, TN, Institute for Software Integrated Systems, Vanderbilt University, 05/2009.
20. Porter, J., P. Volgyesi, N. Kottenstette, H. Nine, G. Karsai, and J. Sztipanovits, "An Experimental Model-Based Rapid Prototyping Environment for High-Confidence Embedded Software", 20th IEEE/IFIP International Symposium on Rapid System Prototyping (RSP'09), Paris, France, 06/2009.
21. Gabor Karsai, Sandeep Neema, David Sharp, Model-driven architecture for embedded software: A synopsis and an example, Science of Computer Programming, Volume 73, Issue 1, 2008, Pages 26-38.

22. Anantha Narayanan, Gabor Karsai, Towards Verifying Model Transformations, Electronic Notes in Theoretical Computer Science, Volume 211, 2008, Pages 191-2008.
23. Narayanan A., Karsai G., "Verifying Model Transformations by Structural Correspondence", Electronic Communications of the EASST, vol. 10, 2008.
24. Karsai, G. and Narayanan, A. 2008. Towards Verification of Model Transformations Via Goal-Directed Certification. In Model-Driven Development of Reliable Automotive Services: Second Automotive Software Workshop, ASWSD 2006, San Diego, Ca, Usa, March 15-17, 2006, Revised Selected Papers, M. Broy, I. H. Krüger, and M. Meisinger, Eds. Lecture Notes In Computer Science, vol. 4922. Springer-Verlag, Berlin, Heidelberg, 67-83.
25. Karsai, G. and Sztipanovits, J. 2008. Model-Integrated Development of Cyber-Physical Systems. In Proceedings of the 6th IFIP WG 10.2 international Workshop on Software Technologies For Embedded and Ubiquitous Systems (Anacapri, Capri Island, Italy, October 01 - 03, 2008). U. Brinkschulte, T. Givargis, and S. Russo, Eds. Lecture Notes In Computer Science, vol. 5287. Springer-Verlag, Berlin, Heidelberg, 46-54.
26. Gray, J., Fisher, K., Consel, C., Karsai, G., Mernik, M., and Tolvanen, J. 2008. DSLs: the good, the bad, and the ugly. In Companion To the 23rd ACM SIGPLAN Conference on Object Oriented Programming Systems Languages and Applications (Nashville, TN, USA, October 19 - 23, 2008). OOPSLA Companion '08. ACM, New York, NY, 791-794.
27. Karsai, G. and Taentzer, G. 2008. Third international workshop on graph and model transformations. In Companion of the 30th international Conference on Software Engineering (Leipzig, Germany, May 10 - 18, 2008). ICSE Companion '08. ACM, New York, NY, 1055-1056.
28. J. Porter, G. Karsai, J. Sztipanovits: Towards a Time-Triggered Schedule Calculation Tool to Support Model-Based Embedded Software Design, ESWeek, 2009.
29. Porter, J., Z. Lattmann, G. Hemingway, N. Mahadevan, S. Neema, H. Nine, N. Kottenstette, P. Volgyesi, G. Karsai, and J. Sztipanovits: The ESMoL Modeling Language and Tools for Synthesizing and Simulating Real-Time Embedded Systems, 15th IEEE Real-Time and Embedded Technology and Applications Symposium, San Francisco, CA, April, 2009.
30. Porter, J., P. Volgyesi, N. Kottenstette, H. Nine, G. Karsai, and J. Sztipanovits: An Experimental Model-Based Rapid Prototyping Environment for High-Confidence Embedded Software, 20th IEEE/IFIP International Symposium on Rapid System Prototyping (RSP'09), Paris, France, June, 2009.
31. J. Skaf and S. Boyd: “Analysis and synthesis of state-feedback controllers with timing jitter,” IEEE Transactions on Automatic Control, 54(3):652-657, March 2009
32. A. Zymnis, S. Boyd, and D. Gorinevsky: ”Relaxed maximum a posteriori fault identification,” Signal Processing, 89(6):989-999, June 2009
33. S. Joshi and S. Boyd: “Sensor selection via convex optimization,” IEEE Transactions on Signal Processing, 57(2):451-462, February 2009
34. A. Magnani and S. Boyd: “Convex piecewise-linear fitting,” Optimization and Engineering, 10(1):1-17, March 2009
35. Y. Wang and S. Boyd: “Performance bounds for linear stochastic control,” Systems and Control Letters, 58(3):178-182, March 2009

36. J. Mattingley and S. Boyd : “Automatic code generation for real-time convex optimization,” To appear as chapter in Convex Optimization in Signal Processing and Communications, Y. Eldar and D. Palomar, Eds., Cambridge University Press, 2009
37. D. Gorinevsky, S.-J. Kim, S. Beard, S. Boyd, and G. Gordon: “Optimal estimation of deterioration from diagnostic image sequence,” IEEE Transactions on Signal Processing, 57(3):1030-1043, March 2009
38. Y. Xu, K.-L. Hsiung, X. Li, I. Nausieda, L. Pileggi, and S. Boyd: “Regular Analog/RF Integrated Circuits Design Using Optimization with Recourse Including Ellipsoidal Uncertainty,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 28(5):623-637, May 2009

39. S. Joshi and S. Boyd: “An Efficient Method for Large-Scale Gate Sizing’” IEEE Transactions on Circuits and Systems I, 55(9):2760-2773, November 2008
40. J. Skaf and S. Boyd: “Nonlinear Q-design for convex stochastic control,” To appear in IEEE Transactions on Automatic Control, 2009
41. R. Panicker, J. Kahn, and S. Boyd: Compensation of multimode fiber dispersion using adaptive optics via convex optimization,” IEEE Journal of Lightwave Technology, May 2008
42. Z. Wang, S. Zheng, Y. Ye and S. Boyd: “Further relaxations of the semidefinite programming approach to sensor network localization,” Siam Journal on Optimization, July 2008
43. D. O’Neill, A. Goldsmith, and S. Boyd: “Optimizing adaptive modulation in wireless networks via utility maximization’” Proc. IEEE International Conf. on Comm., pages 3372-3377, May 2008
44. S.-J. Kim, A. Zymnis, A. Magnani: “Learning the kernel via convex optimization,” Proc. IEEE Internaltional Conf. on Acoustics, Speech, and Signal Processing, pages 1997-2000, April 2008
45. J. Skaf and S. Boyd: “Design of affine controllers via convex optimization,” Submitted to IEEE Transactions on Automatic Control, April 2008
46. K.-L. Hsiung, S.-J. Kim, and S. Boyd: “Tractable approximate robust geometric programming,” Optimization and Engineering, June 2008
47. Y. Wang and S. Boyd: “Fast model predictive control using online optimization,” To appear IEEE Transactions on Control Systems Technology
48. M. Zavlanos, A. Julius, S. Boyd, and G. Pappas: “Identification of stable genetic networks using convex programming,” Proc. American Control Conf., pages 2755-2760, June 2009
49. S. -J. Kim, K. Koh, S. Boyd, and D. Gorinevsky: “l_1 Trend Filtering,” SIAM Review, problems and techniques, May 2009
50. A. Mutapcic, and S. Boyd: “Cutting-set methods for robust convex optimization with pessimizing oracles,” Optimization Methods and Software, June 2009
51. A. Mutapcic, S. Boyd, A. Farjadpour, S. Johnson, and Y. Avniel: “Robust design of slow-light tapers in periodic waveguides,” Engineering Optimization, April 2009

52. J. Ding, J. Sprinkle, S. S. Sastry and C. J. Tomlin: “Reachability analysis for an Automatic Refueling Protocol,” IEEE Conference on Decision and Control, December 2008.
53. S. Forbes. Real-time C Code Generation in Ptolemy II for the Giotto Model of Computation, M.Sc. thesis, EECS Department, University of California, Berkeley, 2009.
54. Ben Lickly, Isaac Liu, Sungjun Kim, Hiren D. Patel, Stephen A. Edwards and Edward A. Lee, "Predictable Programming on a Precision Timed Architecture," in Proceedings of International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), October, 2008.
55. Shanna-Shaye Forbes, Hugo A. Andrade, Hiren Patel, Edward A. Lee, "An Automated Mapping of Timed Functional Specification to A Precision Timed Architecture", in Proceedings of the 12th IEEE International Symposium on Distributed Simulation and Real Time Applications, October, 2008
56. Hiren D. Patel, Ben Lickly, Bas Burgers and Edward A. Lee, "A Timing Requirements-Aware Scratchpad Memory Allocation Scheme for a Precision Timed Architecture," EECS Department, University of California, Berkeley, Technical Report No. UCB/EECS-2008-115, September 12, 2008.
57. Gang Zhou, "Partial Evaluation for Optimized Compilation of Actor-Oriented Models," Ph.D. Dissertation, EECS Department, University of California, Berkeley, Technical Report No. UCB/EECS-2008-53, May 16, 2008.
58. J. Sprinkle, J. M. Eklund, H. Gonzalez, E. I. Grøtli, B. Upcroft, A. Makarenko, W. Uther, M. Moser, R. Fitch, H. Durrant-Whyte and S. S. Sastry. *Model-based design: a report from the trenches of the DARPA Urban Challenge*. Software and Systems Modeling, 2009.
59. H. Gonzalez, E. I. Grøtli, T. R. Templeton, J. O. Biermeyer, J. Sprinkle, and S. Shankar Sastry. *Transitioning Control and Sensing Technologies from Fully-autonomous Driving to Driver Assistance Systems*. Presented at AAET'08.
60. G. M. Hoffmann and C. J. Tomlin, Mobile Sensor Network Control using Mutual Information Methods and Particle Filters, Accepted to appear in the IEEE Transactions on Automatic Control, 2009.
61. H. Huang, G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, Aerodynamics and Control of Autonomous Quadrotor Helicopters in Aggressive Maneuvering, Proceedings of the IEEE Int. Conf. on Robotics and Automation (ICRA), Kobe, Japan, May 2009.
62. G. M. Hoffmann and C. J. Tomlin, Decentralized Cooperative Collision Avoidance for Acceleration Constrained Vehicles, In the Proceedings of the 47th IEEE Conference on Decision and Control, Cancun, Mexico, December 2008.
63. M. P. Vitus, S. L. Waslander, and C. J. Tomlin, Locally optimal decomposition for autonomous obstacle avoidance with the Tunnel-MILP algorithm, In the Proceedings of the 47th IEEE Conference on Decision and Control, Cancun, Mexico, December 2008.
64. G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, Quadrotor Helicopter Trajectory Tracking Control, Proceedings of the AIAA Guidance, Navigation, and Control Conference, August 2008.
65. M. P. Vitus, V. Pradeep, G. M. Hoffmann, S. L. Waslander and C. J. Tomlin, Tunnel-MILP: Path Planning with Sequential Convex Polytopes, Proceedings of the AIAA Guidance, Navigation, and Control Conference, August 2008.

66. M. P. Vitus and C. J. Tomlin, Hierarchical, Hybrid Framework for Collision Avoidance Algorithms in the National Airspace, Proceedings of the AIAA Guidance, Navigation, and Control Conference, August 2008.

6. Interactions/Transitions

6.1 Participation/presentations at meetings, conferences, seminars

1. MURI team attended the bi-weekly MURI telecons.
2. AFOSR Dynamics and Control Program Review, Arlington, VA, August 6, 2009. Janos Sztipanovits: Frameworks and Tools for High-Confidence Design of Adaptive, Distributed Embedded Control Systems: Project Overview
3. HCDDES Review Meeting, October 14, 2008, Berkeley.
   Edward Lee presented "Principled Design of Embedded Software"
   Claire Tomlin and Shanka Sastry presented " Demonstration of the Starmac Experimental Platform and Overview of Hybrid Control Design Challenges"
   Gabor Karsai presented "Model-Based Tool Chain for High Confidence Design"
   Janos Sztipanovits presented "Project Overview"
   Stephen Boyd presented "Robust Control Design"
   Bruce Krogh presented "Model-based Testing and Verification of Embedded System Implementations"
   Nicholas Kottenstette presented "Inertial Control of a Quad-Rotor Helicopter: A Passivity Based Approach"
   Andre Platzer and Edmund Clarke presented: "Saturation-based Scaling Techniques for Symbolic Verification of Hybrid Systems"
4. International Conference on Hybrid Systems Computation and Control 2009, April 14-16, 2009, San Francisco. Alexandre Donze, Bruce H. Krogh:.Parameter Synthesis for Hybrid Systems with an Application to Simulink Models
5. 2008 American Control Conference, Seattle, June 2008. James Kapinski, Bruce H. Krogh, Model checking in-the-loop
6. Workshop on Verification of Numerical Software, San Francisco, April 2009. Bruce H. Krogh: Polyhedral Domains and Widening for Verification of Numerical Programs.
7. IEEE Conference on Decision and Control, Dec. 2008, Cancun, Mexico. Bruce H. Krogh: Performance Bounds on State-Feedback Controllers with Network Delay.
8. IEEE Conference on Decision and Control, Dec. 2008, Cancun, Mexico.. Bruce H. Krogh, Control Software Model Checking Using Bisimulation Functions for Nonlinear Systems.
9. The 6th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS08), September 15—17, 2008, Salzburg, Austria. Bruce H. Krogh: From Analysis to Design
10. 47th IEEE Conference on Decision and Control, December 9-11, Cancun Mexico. Nicholas Kottenstette presented "Wireless control of passive systems subject to actuator constraints"

11. 29th IEEE Real-Time Systems Symposium (RTSS 2008), Barcelona, Spain. Nicholas Kottenstette presented "Passivity-Based Design of Wireless Networked Control Systems for Robustness To Time-Varying Delays
12. Joseph Porter, Graduate student at Vanderbilt visited Prof. Stephen Boyd's Lab for three weeks in May, 2008.
13. Narayanan A., Karsai G., "Verifying Model Transformations by Structural Correspondence", talk given by G. Karsai at GraMoT workshop at the International Conference on Software Engineering, 2008, Leipzig, Germany.
14. Karsai, G. and Sztipanovits, J. 2008. Model-Integrated Development of Cyber-Physical Systems. Talk given by G. Karsai at the 6th IFIP WG 10.2 international Workshop on Software Technologies For Embedded and Ubiquitous Systems (Anacapri, Capri Island, Italy, October 01 - 03, 2008).
15. Model-Integrated Computing: Principles and Examples from Cyber-Physical Systems, talk given by G. Karsai at United Technologies Research Center, Sep 2008.
16. Graham Hemingway, Nicholas Kottenstette, Sandeep Neema, Harmon Nine, Joe Porter, Janos Sztipanovits, and Gabor Karsai: Model-Integrated Toolchain for High Confidence Design, talk given by G. Karsai and demonstration given by J. Porter at the Safe & Secure Systems & Software Symposium, Dayton, OH, June 2009.
17. Janos Sztipanovits: "Model-based design: Challenges and Opportunities," *DATE 2008* , Tutorial on Automation to Realize Embedded Systems From High-Level Functional Models, Munich, Germany , March 10, 2008
18. Sztipanovits, J.: "Convergence: Model-Based Software, Systems and Control Engineering," OOPSLA 2008, Nashville, TN., October 22, 2008
19. Sztipanovits, J.: "Model-based Software Development," (Keynote) Emerging Technology Conference, (ETC08) Huntsville, AL., March 27, 2008
20. Sztipanovits, J.: "Crosscutting CPS Needs in Industry," (Keynote) National Transportation Workshop, Vienna, VA, November 19, 2008
21. Sztipanovits, J.: "High Confidence Design of Embedded Software," University of Vienna, Vienna, Austria May 14, 2008

In addition to the meetings above, we have presented our research results at the following conferences: 2008 IEEE CDC, 2008 AIAA GNC, 2008 CASES, AAET 2008, and the 2008 IEEE International Symposium on Distributed Simulation and Real Time Applications.

We have also presented our results to Boeing, BAE, Thales, BAE Systems, and Lockheed Martin.

6.2 Consultative and advisory functions to other laboratories and agencies, especially Air Force and other DoD laboratories. Provide factual information about the subject matter, institutions, locations, dates, and names(s) of principal individuals involved

1. Janos Sztipanovits:
    a. Study Chair of the AF SAB FY08 Study on "Defending and Operating in a Cyber Contested Environment"
    b. Member of the NASA Advisory Council - Exploration Subcomittee on Avionics, SW and Cybersecurity. 2009-2012

2. Edward A. Lee:
    a. Air Force Research Laboratory, AFRL/RIEA, Rome, NY
    Michael Manno
    michael.manno@rl.af.mil
    (315) 330-7517=20

    The objective of the Extensible Modeling and Analysis Framework (EMAF) effort is tobuild on top of Ptolemy II and adapt Ptolemy II for the rapid construction and configuration of modeling and analysis systems that incorporate disparate technologies. The purpose of this gap-filling project is to develop technologies for future incorporation into large-scale modeling and analysis systems, with specific focuses on scalable algorithm description, composition of heterogeneous components, and synthesis of efficient deployable decision-support systems that exploit multicore and distributed computing platforms.
    In particular, we have applied the code generation infrastructure developed under this MURI to a very large problem consisting of roughly 13000 actors. We were able to reduce the run time from roughly 10 minutes to 3 seconds.

    b. Lockheed Martin Advanced Technology Laboratory
    Trip Denton
    ldenton@atl.lmco.com
    3 Executive Campus, 6th Floor;Cherry Hill, NJ, 08002, USA
    Work: 856 792-9071 fax:856 792-9925

    NAOMI Project (http://chess.eecs.berkeley.edu/naomi)
    (Also participating are Vanderbilt and UIUC)
    The purpose of the NAOMI project is to allow disparate modeling tools to be used to ether by tracking model changes within each system where a particular tool owns attributes of the overall design and provides attribute changes to other tools. The NAOMI project may result in useful technology that will allow easier collaboration on this MURI project. This project is using pedestrian/automobile traffic lights as a design driver. We have integrated Ptolemy II to the Naomi framework, which allows different tools to own attributes and update other tools when changes occur to those attributes.
    We have transferred models that use graph transformation and event relationship graphs.

    c. The US Army Research Laboratory
    Jeff DeHart, jdehart@arl.army.mil
    Scalable Composition of Systems (SCOS)
    http://chess.eecs.berkeley.edu/scos

    The objective of the SCOS research project is to provide scalable techniques for the composition of subsystems in a system-of-systems (SoS) framework for large,

complex applications such as FCS.

SCOS has synergy with this MURI project in that it deals with large systems. In particular:
- we are using the EmbeddedCActor to wrap legacy C code
- we are collaborating on work on the Kepler Project
- we are using Graph Transformations on models

3. Bruce Krogh
    a. National Science Foundation.
    Helen Gill hgill@nsf.gov
    Contributed to the development of the NSF Soliciation for Cyber-Physical Systems.
    b. Lockheed Martin Advance Development Projects (ADP)
    Peter Stanfill peter.o.stanfill@lmco.com
    Consultant to the LM team in the AFRL MCAR program.

6.3 Technology Assists, Transitions, and Transfers.

1. 8th Biennial Ptolemy Miniconference, Thursday, April 16, 2009 in Berkeley, California

2. Key components of Vanderbilt's MIC tool suite (GME, GReAT, UDM) had one major release in 2009. The released tools are available through the ESCHER and ISIS download sites
3. Vanderbilt continued working with GM, Raytheon, Lockheed Martin, Boeing and BAE Systems research groups on transitioning model-based design technologies into programs.
4. Vanderbilt continued working with Boeing's FCS program on applying the MIC tools for precise architecture modeling and systems integration

6.4 New discoveries, inventions, or patent disclosures.

None.

6.5 Honors and Awards

1. Edmund M. Clarke
    a. Best Paper Award – Edmund M. Clarke, Alexandre Donzé, Axel Legay: Statistical Model Checking of Mixed-Analog Circuits with an Application to a Third Order Delta-Sigma Modulator. Haifa Verification Conference, October 27-30, 2008, Haifa, Israel

b. Strachey Lecture – Edmund M. Clarke. My 27-year Quest to Overcome the State Explosion Problem. Oxford University Computing Laboratory, May 12, 2009, Oxford, UK
c. Technion CS Distinguished Lectures – Edmund M. Clarke. Technion - Israel Institute of Technology, May 17-26, 2009, Haifa, Israel.
d. Keynote Speaker – Edmund M. Clarke\
    i. Model Checking – My 27-year Quest to Overcome the State Explosion Problem. NASA Formal Methods Symposium, April 6-8, 2009, Moffett Field, CA
    ii. Model Checking - My 27-Year Quest to Overcome the State Explosion Problem. LPAR 2008, November 22-27, 2008, Doha, Qatar
    iii. BMC: Before Model Checking. CAV 2008, July 7-14, 2008, Princeton, NJ
    iv. Model Checking. DAC 2008, June 8-13, 2008, Anaheim, CA
    v. U.S. Department of Defense Workshop on Satisfiability, March 3-5, 2009, Baltimore, MD

2. Claire Tomlin:
    a. Chancellor's Professorship of EECS, UC Berkeley (2007-2010)
    b. Tage Erlander Guest Professorship, Swedish Research Council, 2009.
    c. Engineering Alumni Achievement Medal, University of Waterloo, 2007-2008.
3. Shankar Sastry:
    a. Appointed Dean of Engineering, UC Berkeley, July 2007 -
4. Janos Sztipanovits:
    a. Keynotes:
        i. "Convergence: Model-Based Software, Systems and Control Engineering," OOPSLA 2008, Nashville TN., October 22, 200
        ii. "Model-based Software Development," Emerging Technology Conference, (ETC08) Huntsville, AL., March 27, 2008
        iii. "Crosscutting CPS Needs in Industry," National Transportation Workshop, Vienna, VA, November 19, 2008
    b. Georgia Tech ECE Distinguished Lectures: "Three Problems of Model-based Design," Atlanta, GA, November, 2008
    c. Appointment: NASA Advisory Council - Exploration Subcomittee on Avionics, SW and Cybersecurity. 2009-2012